January 2021

SHARING BLUETOOTH PAIRING BETWEEN MULTIPLE HOST DEVICES

Douglas Stockwell

SHARING BLUETOOTH PAIRING BETWEEN MULTIPLE HOST DEVICES

ABSTRACT

When initially pairing devices using wireless communication technologies (e.g., Bluetooth®, Bluetooth Low Energy (BLE), WiFi®, etc.), the devices may exchange a common secret key to enable the device to automatically reestablish the connection in the future. Rather than requiring a new common secret key to be exchanged, techniques of this disclosure enable different host devices (e.g., phones, computers, watches, etc.) to share a common secret key for each client device (e.g., headphones, printers, keyboards, mice, etc.). A virtualized pairing service, which may be provided by a cloud-based or other computing system, may receive common secret keys for a particular user account each time the user performs an initial pairing of a client device with a host device. Other host devices associated with the user account may download (automatically or manually) the new common secret key from the virtualized pairing service. After the initial pairing of that client device, the user may wish to pair the client device with a different host device. Rather than having to repeat the pairing process and generate a new secret key, the different host device may use the common secret key for the client device downloaded from the virtualized pairing service to establish a connection as if the different host device and the client device were already paired. Thus, rather than performing the typical pairing process for each new host device the user may wish to connect to the client device, techniques of this disclosure enable the various devices associated with the user to reuse information (e.g., common secret keys) generated during an initial pairing with one device, which may simplify the subsequent connection process for new host and/or client devices.

DESCRIPTION

In general, when initially pairing devices using wireless communication technologies (e.g., Bluetooth®, Bluetooth Low Energy (BLE), WiFi®, etc.), the devices may exchange a common secret key to enable the device to automatically reestablish the connection in the future. Rather than requiring a new common secret key to be exchanged, techniques of this disclosure enable different host devices (e.g., phones, computers, watches, etc.) to share a common secret key for each client device (e.g., headphones, printers, keyboards, mice, etc.).

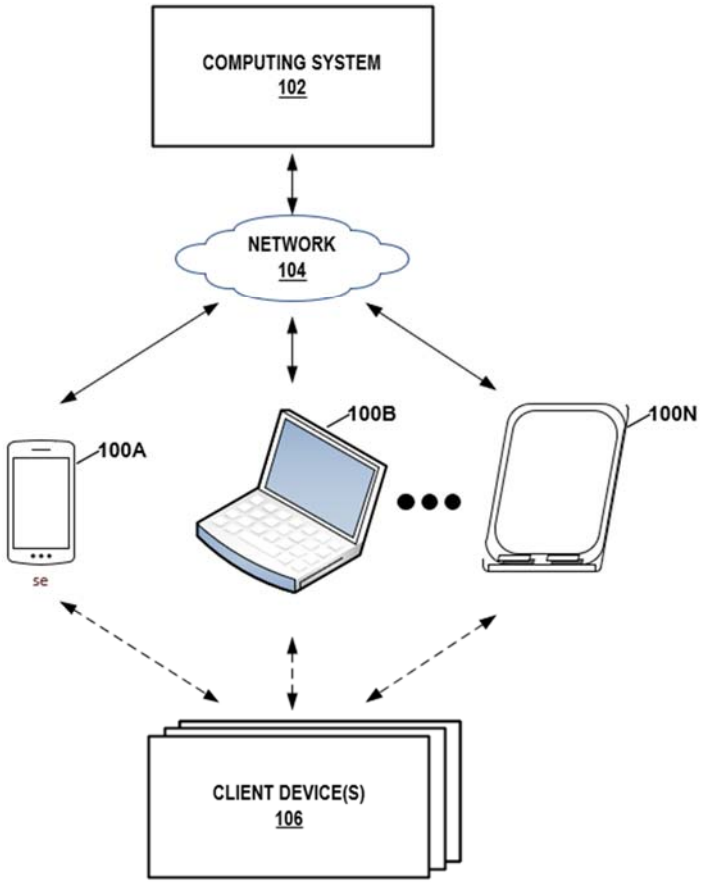

FIG. 1

FIG. 1 is a conceptual diagram illustrating a computing system 102 configured to enable pairing between one or more client devices 106 and one or more host devices 100A-100N (collectively, “host devices 100”). Client devices 106 may be any headphone, printer, keyboard,

mouse, or another client device that may pair with host devices 100 using wireless communication technologies. Host devices 100 may be any mobile or non-mobile computing device, such as a cellular phone, a smartphone, a personal digital assistant (PDA), a desktop computer, a laptop computer, a tablet computer, a portable gaming device, a portable media player, an e-book reader, a watch (including a so-called smartwatch), an add-on device (such as a casting device), smart glasses, a gaming controller, or another type of computing device that may pair with client devices 106 using wireless communication technologies.

Host devices 100 may include communication ("COMM") components. COMM components may receive and transmit various types of information, such as pairing details (e.g., the common secret key for each client device 106, such as headphones, printers, keyboards, mice, etc.). COMM components may include wireless communication devices capable of transmitting and/or receiving communication signals (e.g., via a network 104, to and from client devices 106, etc.), such as a cellular radio, a 3G radio, a 4G radio, a 5G radio, a Bluetooth® radio (or any other personal area network (PAN) radio), a near-field communication (NFC) radio, or a WiFi® radio (or any other wireless local area network (WLAN) radio).

Computing system 102 may be any suitable remote computing system, such as one or more desktop computers, laptop computers, mainframes, servers, cloud computing systems, virtual machines, and/or the like capable of sending and receiving information via network 104. In some examples, computing system 102 may represent a cloud computing system that provides one or more services (e.g., cloud services) via network 104. That is, in some examples, computing system 102 may be a distributed computing system.

One or more computing devices, such as host device 100A, may access the services provided by the cloud by communicating with computing system 102 via network 104. Network

104 may include a wide-area network (WAN) such as the Internet, a local-area network (LAN), a PAN (e.g., Bluetooth®), an enterprise network, a wireless network, a cellular network, a telephone network, a Metropolitan area network (e.g., WiFi®, WAN, worldwide interoperability for microwave access (WiMAX), etc.), one or more other types of networks, or a combination of two or more different types of networks (e.g., a combination of a cellular network and the Internet).

In accordance with techniques of this disclosure, computing system 102 may enable different host devices 100 to share a common secret key for each client device 106. For example, computing system 102 may be a cloud server that allows a user to create and manage a user account. Computing system may receive and store information about the user's host devices 100 and client devices 106 that the user associates with the user account. For example, computing system 102 may receive a common secret key for client devices 106 each time the user performs an initial pairing between client device 106 and one of host devices 100. Computing system 102 may then store the pairing details (e.g., the common secret key for each client device 106) for client devices 106 in a pairing details repository. In addition, computing system 102 may store information about host devices 100, such as a device name, an identification number, a device priority, and/or the like.

Computing system 102 may share the common secret key for client devices 106 with one or more host devices 100 associated with the particular user account. That is, host devices 100 may communicate with computing system 102 via network 104 to receive the pairing details for the client devices. For example, host device 100A, which is a smartphone in the example of FIG. 1, may download the pairing details for each of client devices 106 from computing system 102 from a virtualized pairing service provided by computing system 102. The smartphone may then

use the downloaded pairing details to pair with one or more client devices 106, such as a wireless keyboard. As such, rather than exchange a new common secret key, a smartphone may download a common secret key for the wireless keyboard from the virtualized pairing service to pair with the wireless keyboard.

In some examples, in response to determining that one or more common secret keys for client devices 106 are not stored at host devices 100, host devices 100 may attempt to download the one or more common secret keys from the virtualized pairing service. For example, if host device 100A (e.g., a smartphone) is attempting to pair with the wireless keyboard and the storage device of the smartphone stores the common secret key for the wireless keyboard, then the smartphone may pair with the keyboard using the common secret key in the storage device of the smartphone without downloading any common secret keys from the virtualized pairing service. However, if the storage device of the smartphone does not store the common secret key for the wireless keyboard, then the smartphone may download the common secret key for the wireless keyboard from the virtualized pairing service and then pair with the wireless keyboard using the downloaded common secret key.

In some examples, connection with client devices 106 may be handed-off from one host device 100 (e.g., host device 100A) to another of host devices 100 (e.g., host device 100B). For example, host device 100A may initially be connected to a wireless keyboard. Responsive to receiving an input (e.g., user input received by host device 100B) or the satisfaction of one or more conditions (e.g., proximity, priority, etc.), host device 100A may terminate connection with the keyboard, and host device 100B may connect with the wireless keyboard.

In some examples, host devices 100 may present (e.g., via a display) a GUI allowing the user to hand-off a connection with client devices 106 from one host device to another host

device. For example, a presence-sensitive display of host device 100B may present a graphical element (e.g., a button, an icon, etc.) associated with functionality for handing-off client device 106 from host device 100A to host device 100B. In this way, the user may manually connect host device 100B to client device 106.

In some examples, host devices 100 may automatically hand-off connections based on the proximity of host devices 100 to one or more of client devices 106. For example, the one of host devices 100 (e.g., host device 100B) that is physically closest to (i.e., has the strongest wireless communication signal strength with) one of client devices 106 (e.g., wireless headphones) may automatically connect to the wireless headphones. Thus, if the host device 100B is initially the closest to the wireless headphones, client device 106 may automatically initially pair with host device 100B. Responsive to host device 100B and the wireless headphones moving physically apart such that host device 100A is now the host device closest to the wireless headphones, host device 100B may automatically terminate connection with the wireless headphones, and host device 100A may automatically pair with the wireless headphones. In various instances, host device 100A or 100B may output an alert to notify the user that the wireless headphones will switch to being paired with host device 100A if the user does not provide any user input to prevent the switch, does not move physically closer to the wireless headphones, and/or take some other action.

In some examples, host devices 100 may automatically hand-off connections based on a priority ranking of host devices 100 that are near client devices 106. For example, the one of host devices 100 (e.g., host device 100B) with the highest device priority within a predetermined range (e.g., based on communication signal strength between) of one of client devices 106 (e.g., wireless headphones) may automatically connect to the wireless headphones. Thus, if host device

100B is the highest device priority within the predetermined range (e.g., 50 feet) of the wireless headphones, the wireless headphones may initially automatically pair with host device 100B. Responsive to host device 100B being outside of the predetermined range of the wireless headphones, such that host device 100A is now the host device with the highest device priority within the predetermined range of the wireless headphones, host device 100B may automatically terminate connection with the wireless headphones, and host device 100A may automatically pair with the wireless headphones. As described above, in various instances, host device 100A or 100B may output an alert to notify the user that the wireless headphones will switch to being paired with host device 100A if the user does not provide any user input to prevent the switch, does not move physically closer to the wireless headphones, and/or take some other action.

One or more advantages of the techniques described in this disclosure include improving the user experience by reducing the amount of user interaction required to hand-off a connection with a client device from one host device to another host device. Another advantage is that multiple host devices associated with a particular user account may download the pairing details from the virtualized pairing service provided by the computing system. Thus, rather than performing the typical pairing process for each host device the user may wish to connect to the client device, techniques of this disclosure enable the various host devices associated with the particular user account to reuse information generated during an initial pairing with one device, which may simplify the subsequent connection process for new host and/or client devices.

References

1. US Patent Application Publication No. US20150351142A1
2. US Patent Application Publication No. US20160360341A1
3. US Patent Application Publication No. US20170005820A1